Cofc

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Philbrick et al.		Patent No.:	7,076,568
Issue Date:	July 11, 2006	Ser. No:	09/802,551
Filing Date:	March 9, 2001	Examiner:	Jungwon Chang
Atty. Docket No:	ALA-012	GAU:	2154

For: INTELLIGENT NETWORK STORAGE INTERFACE SYSTEM

November 6, 2006

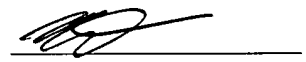
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313

Replacement Request for Certificate of Correction under 37 C.F.R. §1.322

Sir:

Please consider this Replacement Request for Certificate of Correction as the Applicants' inadvertently identified an incorrect patent number in the original Request for Certificate of Correction, dated September 28, 2006.. The Applicants have corrected the patent number on this Replacement Request. Therefore, enclosed is a Certificate of Correction, form PTO/SB/44, for the above-referenced patent. Also enclosed is a copy of an Amendment, dated February 27, 2006 and filed March 2, 2006, which shows, on page 4, line 14 that the mistakes appear to have been made by the Patent Office. Therefore, no fee is required. Please issue the enclosed Certificate of Correction.

Respectfully submitted,

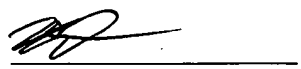


Mark Lauer
Reg. No. 36,578
6601 Koll Center Parkway
Suite 245
Pleasanton, CA 94566
Tel: (925) 484-9295
Fax: (925) 484-9291

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313, on November 6, 2006.

Date: 11-06-06



Mark Lauer

PTO/SB/44 (04-05)
Approved for use through 04/30/2007. OMB 0651-0033
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

(Also Form PTO-1050)

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 7,076,568 B2

APPLICATION NO.: 09/802,551

ISSUE DATE : July 11, 2006

INVENTOR(S) : Clive M. Philbrick et al.

It is certified that an error appears or errors appear in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

In the Claims:
Claim 12, Column 38, lines 48-51, the text "12. The apparatus of claim 11, wherein said host computer contains a file system and said interface memory includes a file cache adapted to store said data, wherein said file system manages storage of said data in said file cache." should be deleted.

MAILING ADDRESS OF SENDER (Please do not use customer number below):
Mark Lauer
6601 Koll Center Parkway, Suite 245
Pleasanton, CA 94566

NOV 16 2006

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Clive M. Philbrick et al.	Ser. No:	09/802,551
Filing Date:	March 9, 2001	Examiner:	Jungwon Chang
Atty. Docket No:	ALA-012	GAU:	2154
Assignee:	Alacritech, Inc.		

For: INTELLIGENT NETWORK STORAGE INTERFACE SYSTEM

February 27, 2006

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Second Amendment

In response to an Office Action mailed November 16, 2005, please enter the following Amendment to the Claims and consider the following Remarks.

Amendment to the Claims

1. (Previously presented) An apparatus for transferring information between a network and a storage device, the apparatus comprising:

a host computer having a CPU operating a file system and a host memory connected to said CPU by a host bus, and

an interface device coupled to said host computer, to the network and to the storage device, said interface device including an interface memory containing an interface file cache adapted to store data that is communicated between the network and the storage device under control of said file system,

wherein said host computer is configured to designate a User Datagram Protocol socket that is accessible by said interface device, and said interface device is configured to communicate said data between the network and the file cache according to said User Datagram Protocol socket.

2. (Original) The apparatus of claim 1, wherein said host computer is configured to create an application layer header that is accessible by said interface device, and said interface device is configured to prepend said application layer header to said data.

3. (Original) The apparatus of claim 1, wherein said host computer is configured to create a Realtime Transport Protocol header that is accessible by said interface device, and said interface device is configured to prepend said Realtime Transport Protocol header to said data.

4. (Previously presented) The apparatus of claim 1, wherein said data is stored with an associated User Datagram Protocol header, and said interface device includes a mechanism configured to process said User Datagram Protocol header.

5. (Previously presented) The apparatus of claim 1, wherein said data is prepended with a User Datagram Protocol header by said interface device to create a User Datagram Protocol datagram, and said interface device includes a mechanism configured to divide said datagram into plural fragments.

6. (Previously presented) The apparatus of claim 1, wherein said data is disposed in plural fragments, and said interface device includes a mechanism configured to concatenate said fragments corresponding to a User Datagram Protocol header.

7. (Original) The apparatus of claim 1, wherein said data does not enter said host computer.

8. (Original) The apparatus of claim 1, wherein said data includes audio data.

9. (Original) The apparatus of claim 1, wherein said data includes video data.

10. (Original) The apparatus of claim 1, wherein said data is a part of a realtime communication.

11. (Currently amended) An apparatus for transferring information between a network and a peripheral device, the apparatus comprising:

a host computer having a processor running a file system and connected to a host memory by a host memory bus, said host memory containing an application operable by the processor to designate a User Datagram Protocol socket, and

an interface device connected to said host computer and coupled between the network and the peripheral device, said interface device including an interface ~~memory-adapted~~ file cache managed by said file system to store data corresponding to said User Datagram Protocol socket and a mechanism configured to associate said data with a User Datagram Protocol header corresponding to said User Datagram Protocol socket such that said data is communicated between the network and the peripheral device without encountering said host computer.

12. (Canceled) The apparatus of claim 11, wherein said host computer contains a file system and said interface memory includes a file cache adapted to store said data, wherein said file system manages storage of said data in said file cache.

13. (Previously presented) The apparatus of claim 11, wherein said data travels over the network in at least one packet containing a User Datagram Protocol header, and said interface device includes circuitry configured to process said User Datagram Protocol header.

14. (Previously presented) The apparatus of claim 11, wherein said data travels over the network in plural fragments corresponding to a User Datagram Protocol header, and said interface device is configured to concatenate said data with said User Datagram Protocol header.

15. (Original) The apparatus of claim 11, wherein said host computer is configured to create a Realtime Transport Protocol header that is accessible by said interface device, and said interface device is configured to prepend said Realtime Transport Protocol header to said data.

16. (Original) The apparatus of claim 11, wherein said data includes audio data.

17. (Original) The apparatus of claim 11, wherein said data includes video data.

18. (Original) The apparatus of claim 11, wherein said data is a part of a realtime communication over the network.

19-20. Canceled

21. (Previously presented) An apparatus for transferring information between a network and a storage device, the apparatus comprising:

a host computer having a CPU operating a file system and a host memory connected to said CPU by a host bus, and

an interface device coupled to said host computer, to the network and to the storage device, said interface device including an interface memory containing an interface file cache adapted to store data that is communicated between the network and the storage device under control of said file system,

wherein said host computer is configured to designate a User Datagram Protocol socket that is accessible by said interface device, and said interface device has means for communicating said data between the network and the file cache according to said User Datagram Protocol socket.

22. (Previously presented) The apparatus of claim 21, wherein said host computer is configured to create an application layer header that is accessible by said interface device, and said interface device is configured to prepend said application layer header to said data.

23-25. New

Remarks

I. Double Patenting

Claims 1-18, 21 and 22 stand rejected under the judicially created doctrine of obviousness type double patenting as being unpatentable over claims 1-24 of U.S. Patent No. 6,807,581.

Applicants respectfully disagree with this rejection. For instance, the Office Action states: "Changing the name will not serve as a basis for patentability." Applicants respectfully assert, however, that claims are made of words, such as names for elements or limitations. The logical extension of the Office Action reasoning would arbitrarily preclude patentability of any new claims that have different words, such as names, than issued claims.

Although applicants disagree with this rejection, in an effort to expedite the already extended prosecution of this application, applicants are submitting along with this response a Terminal Disclaimer over U.S. Patent No. 6,807,581.

II. 35 U.S.C. §103

A. The Rejection of Claims 1, 3-10, 12 and 21

Claims 1, 3-10, 12 and 21 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,913,028 to Wang et al. in view of U.S. Patent No. 5,848,293 to Gentry and U.S. Patent No. 6,385,647 to Willis et al. The Office Action states:

> Claims 1, 3-10, 12 and 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Wang et al. (US 5,913,028) (hereinafter Wang) in view of Gentry (US 5,848,293) (hereinafter Gentry) and further in view of Willis et al. (US 6,385,647) hereinafter Willis.
>
> As for claims 1 and 21, Wang discloses an apparatus for transferring information between a network and a storage device, the apparatus comprising:
>
> a host computer having a CPU (CPU 24, Fig. 2) operating a file system (file system 27, direct file system 28, and peer I/O manager 26, Fig. 3) and a host memory (memory 32, Fig. 2) connected to said CPU by a host bus (Figs. 2 and 3), and

an interface device (40, Fig. 2) coupled to said host computer, to the network and to the storage device, said interface device including an interface memory (local memory 44, Fig. 3) adapted to store data that is communicated between the network and the storage device under control of said file system (Network I/O device, Fig. 3; col. 3, lines 31-52; col.4, line 50 – col. 5, line 5; col. 6, line 66 – col. 7, line 7; col. 8, lines 1-11; uses low level file system primitives such as the Netware Direct File System 28 to issue direct read requests to the storage I/O Device 40, so that data transfers are accomplished,

wherein said host computer is configured to designate *a socket* that is accessible by said interface device, and said interface device is configured to communicate said data between the network and the file cache according to said *socket* (Fig. 3; col. 4, line 38 – col. 5, line 5; col. 11, lines 1-11).

Wang discloses network device including an interface memory (local memory; Fig. 3; 44, Fig. 5). However, Wang does not explicitly disclose the interface memory contains an interface file cache. Gentry discloses the interface memory contains an interface file cache (Figs. 3, 6; col. 1, lines 39-67; col. 2, lines 38-57, col. 6, line 42 – col. 7, line 12; col. 7, line 40 – col. 8, line 50). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Wang and Gentry because Gentry's interface file cache would operate fast to transfer data between host computer and the other computers in the network (Gentry, col. 4, lines 19-36).

Wang does not explicitly teach that the socket may be a Uniform Datagram Protocol (UDP) socket. Willis teaches UDP socket (UDP socket 5A30, fig. 5; UDP socket 6A20, fig. 6; col. 14, lines 12-67). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Wang and Willis because using the Willis' UDP socket would efficiently transmit data packet to intended destination.

B. Applicants' Response to the Rejection of Claims 1, 3-10, 12 and 21

1. Wang's Peer I/O Manager is not a File System

Initially, applicants respectfully disagree with the Office Action assertion that: "Wang discloses ... a file system (... peer I/O manager 26, Fig. 3)..." For example, in contrast to the Office Action allegation that "peer I/O manager 26, Fig. 3" is "a file system," Wang states in column 11, lines 19-21:

The Peer I/O Manager, like NCP, is a ***protocol layer*** that accepts data packets addressed to ***its own unique Socket ID*** in a file server. The client redirector extension is a client-based software protocol layer that

addresses file requests to the Peer I/O Manager socket ID in the file server. (emphasis added)

According to this passage of Wang, the "Peer I/O Manager" of Wang is a network protocol layer, and is not "a file system" as alleged by the Office Action. Moreover, Wang teaches that the "Peer I/O manager" has "its own unique Socket ID," as opposed to being "a file system" as alleged by the Office Action. Applicants respectfully assert that "a file system" would not have "its own unique Socket ID," but would instead be able to manage data for various applications.

This differentiation between the "Peer I/O Manager" and the "file system" of Wang is further described in column 11, lines 26-36 of that reference, which states:

> Thus, Peer I/O manager services co-exist with traditional file server functions. the NCP protocol facilitates file open/close/read/write operations between traditional Network Attached Clients and file servers. NCP facilitates file read/write requests through Planar-board Centric I/O data transfers, i.e. all data moves through the File Cache in Planar-board memory. Peer I/O Manager, however, facilitates file read/write requests through Peer I/O data transfers, i.e., all data moves directly from Storage I/O device Local Memory to Network I/O Device Local Memory.

In other words, peer I/O manager services coexist with file server functions rather than acting as a file system.

2. Gentry does not Teach or Suggest an Interface File Cache

The Office Action acknowledges that Wang does not disclose an interface memory containing an interface file cache. The Office Action asserts, however, that: "Gentry discloses ... an interface file cache (Figs. 3, 6; col. 1, lines 39-67; col. 2, lines 38-57, col. 6, line 42 – col. 7, line 12; col. 7, line 40 – col. 8, line 50)." Applicants respectfully disagree with this Office Action assertion. Applicants respectfully submit that Gentry does not teach or suggest an "interface file cache" as recited in claim 1, but rather a cache of network channel state entries. For example, Gentry teaches, in column 2, lines 39-44:

> In one embodiment the receiving device is a network interface device which comprises ***a cache memory which stores entries indicative of active channels*** used for data transfer between the host computer system and a network. (emphasis added)

See also column 1, lines 41-43 ("...the state of the cache may change to reflect channel activity. If the state of the cache changes..."); column 1, lines 49-50 ("...activity on the network may cause the channel states, stored in the cache, to change..."); column 1, 55-67 ("...the device responds by performing a form of lookup to determine the physical location of the entry in the cache corresponding to the channel ..."); column 3, lines 6-9 ("...the circuitry performs a lookup to the cache which stores the corresponding physical location of the VCI and certain status bits such as the state of the channel ..."); column 7, lines 8-10 ("A cache is utilized to store a subset of the different channel states, for example, the cache is large enough to store 128 states."); column 8, lines 9-14 ("The VCI ID is used to index into a table to determine whether the information for a particular channel is currently located in the cache 330 (FIG. 3). Referring to FIG. 6, the VCI map 606 identifies whether the VCI is currently cached, item 615, and if it is cached, the location in the cache, item 610."); column 8, lines 25-27 ("When the channel information is removed from the cache 670 it is stored in slower memory 680, which provides storage of channel information for all 1024 channels"); column 8, lines 32-34 ("When a new channel state needs to be brought into the cache it replaces the current least recently used entry...").

In other words, Gentry does not teach or suggest an interface file cache.

3. Wang and Gentry do not Teach or Suggest an Interface File Cache that is Under Control of a File System

Assuming arguendo that Wang and Gentry would have been combined as proposed by the Office Action, Wang and Gentry do not together teach or suggest an "interface file cache," as recited in claim 1, which is "under control of said file system." As noted above, neither of those references discloses an "interface file cache," and Wang's "Peer I/O Manager" is differentiated from the "file system" by that reference. *Assuming arguendo* that Willis would have been combined with Wang and Gentry, as proposed by the Office Action, does not remedy these deficiencies.

Having an "interface file cache" that is "under control of said file system," as recited in claim 1, has advantages in unifying organization and control of files of the

"interface file cache" with other files organized and controlled by the file system. Wang and Gentry do not individually or together teach or suggest such advantages.

4. Wang and Willis would not have been Combined by One of Ordinary Skill in the Art as Proposed by the Office Action

Applicants respectfully assert that Wang's disclosure of data transfer between Wang's "storage I/O device" and "network I/O device" would be destroyed if combined with Willis' UDP implementation as proposed by the Office Action. This is because Wang's "Peer I/O Manager," which appears essential to data transfer between Wang's "storage I/O device" and "network I/O device" is a protocol layer with a "unique Socket ID" that would be rendered ineffectual if replaced with Willis' UDP socket ID as proposed by the Office Action. As stated in column 11, lines 6-21 of Wang, which is cited in the Office Action for teaching a "*socket*":

> The IPX software layer acts much like a mailman. It delivers packets to higher layer protocols through a post office box-like scheme. The IPX layer uses the ***Destination Socket*** in the IPX Header field of each incoming packet as the postal address that associates incoming data packets with specific higher layer protocols. Protocols register with IPX to receive packets addressed to specific ***Socket IDs***.
>
> FIG. 7, illustrates the NetWare™ Core Protocol (NCP) as a software layer above IPX. IPX delivers data packets addressed to ***Socket number 451 to NCP***. Traditionally, NetWare™ client-based redirectors send file requests to ***the NCP socket ID*** in the file server.
>
> ***The Peer I/O Manager***, like NCP, is a ***protocol layer*** that accepts data packets addressed to ***its own unique Socket ID*** in a file server. The client redirector extension is a client-based software protocol layer that addresses file requests to the Peer I/O Manager socket ID in the file server. (emphasis added)

As further stated in column 4, lines 1-5 of Wang:

> The Peer I/O Manager implementation provides the program control necessary for coordinating and initiating peer I/O transfers directly from the storage I/O device to the network I/O device.

In other words, "using the Willis' UDP socket" in place of the "the Peer I/O Manager" as proposed by the Office Action would appear to destroy "the program control necessary for coordinating and initiating peer I/O transfers directly from the storage I/O device to the network I/O device." Applicants respectfully assert that for this

reason one of ordinary skill in the art would not have combined Wang and Willis as proposed by the Office Action.

5. Wang and Gentry and Willis do not Teach or Suggest that said Host Computer is Configured to Designate a User Datagram Protocol Socket that is Accessible by said Interface Device, and said Interface Device is Configured to Communicate said Data between the Network and the File Cache According to said User Datagram Protocol Socket

Assuming arguendo that Wang and Gentry and Willis would have been combined as proposed by the Office Action, Wang and Gentry and Willis do not together teach or suggest that "said host computer is configured to designate a User Datagram Protocol socket that is accessible by said interface device," as recited in claim 1, and that "said interface device is configured to communicate said data between the network and the file cache according to said User Datagram Protocol socket," as further recited in claim 1.

As mentioned above, neither Wang nor Gentry nor their proposed combination teach or suggest an "interface file cache" that is "under control of said file system," as recited in claim 1. Willis also does not teach or suggest such an "interface file cache," or even an interface device. *Assuming arguendo* that Wang, Gentry and Willis would have been combined as proposed by the Office Action, there is no teaching or suggestion in any of those references of a "host computer" that "is configured to designate a User Datagram Protocol socket that is accessible by said interface device," as recited in claim 1. Applicants note that the Office Action does not allege that Wang, Gentry and Willis teach or suggest a "host computer" that "is configured to designate a User Datagram Protocol socket that is accessible by said interface device," as recited in claim 1.

Moreover, *asssuming arguendo* that Wang, Gentry and Willis would have been combined as proposed by the Office Action, there is no teaching or suggestion in any of those references that "said interface device is configured to communicate said data between the network and the file cache according to said User Datagram Protocol socket," as further recited in claim 1. Applicants note that the Office Action also does not allege that Wang, Gentry and Willis teach or suggest an "interface device" that "is configured to communicate said data between the network and the file cache according to said User Datagram Protocol socket," as recited in claim 1.

For all the above reasons, applicants respectfully submit that claim 1 and all the dependent claims that reference claim 1 are nonobvious over the Office Action's proposed combination of Wang, Gentry and Willis.

5. Claim 21

Regarding claim 21, the Office Action does not differentiate claim 21 from claim 1, despite the "means plus function" limitation of claim 21. That is, the Office Action does not compare the structure disclosed in the specification of the present application and corresponding to the "means for communicating said data between the network and the file cache according to said User Datagram Protocol socket" limitation of claim 21 with that of the cited references. For this reason the Office Action has not presented a *prima facie* rejection of claim 21.

6. Claim 3

Regarding claim 3, the Office Action states:

> As for claim 3, Wang does not explicitly disclose the use of Realtime Transport Protocol (RTP) headers. Willis teaches creating RTP headers and prepending the header to the data for transmission over the network (column 11, lines 2-22). It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Wang and Willis because Willis' RTP protocol would improve quality of service by supporting the transmission and reception of real-time multimedia (Willis, col. 2, lines 26-36).

Claim 3, however, recites:

> The apparatus of claim 1, wherein said host computer is configured to create a Realtime Transport Protocol header that is accessible by said interface device, and said interface device is configured to prepend said Realtime Transport Protocol header to said data.

Neither Willis nor Wang teaches or suggests that "said host computer is configured to create a Realtime Transport Protocol header" and that "said interface device is configured to prepend said Realtime Transport Protocol header to said data." Applicants note that the Office Action does not allege that its proposed combination of Willis and Wang teaches or suggests that "said host computer is configured to create a

Realtime Transport Protocol header" and that "said interface device is configured to prepend said Realtime Transport Protocol header to said data." Applicants respectfully assert that, *assuming arguendo* that one of ordinary skill in the art would have combined Wang and Willis as proposed by the Office Action, one of such skill may have instead have used a host computer to prepend a RTP header, or used an interface device to create a RTP header, in contrast to claim 3. For these additional reasons, claim 3 is nonobvious over the references cited in the Office Action.

7. Claim 4

Regarding claims 4, 5 and 6, the Office Action states:

> As for claims 4, 5 and 6, Wang does not explicitly disclose the use of UDP headers. Willis teaches that UDP, by definition, prepends data with UDP headers, wherein the data is further divided into plural fragments which are concatenated corresponding to the UDP header (col. 10, line 40 – col. 11, line 22; col. 14, lines 12-67; col. 15, lines 34-46). It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Wang by using UDP headers and dividing the data into plural fragments, in order to efficiently transfer data over a network, as taught by Willis above.

Claim 4, however, recites:

> The apparatus of claim 1, wherein said data is stored with an associated User Datagram Protocol header, and said interface device includes a mechanism configured to process said User Datagram Protocol header.

Applicants respectfully assert that neither Willis nor Wang teaches or suggests that "said interface device includes a mechanism configured to process said User Datagram Protocol header." Appellants respectfully assert that it is not trivial, and would not have been obvious to one of ordinary skill in the art, to modify Wang by adding transport layer protocol processing to its "network I/O device." Appellants respectfully assert that, *assuming arguendo* that one of ordinary skill in the art would have combined Wang and Willis as proposed by the Office Action, one of such skill may have used "planar board 20 having main CPU(s) 24 and main dynamic memory 22" of Wang to process a UDP header, instead of a "network I/O device," in contrast to claim 4. For

these additional reasons, claim 4 is nonobvious over the references as proposedly combined by the Office Action.

8. Claim 5

Regarding claim 5, the rejection of claims 4, 5 and 6 is shown above. Claim 5, however, recites:

> The apparatus of claim 1, wherein said data is prepended with a User Datagram Protocol header by said interface device to create a User Datagram Protocol datagram, and said interface device includes a mechanism configured to divide said datagram into plural fragments.

It is clear from Wang that the "network I/O device" only processes the network layer protocol, not transport layer protocols such as UDP, as discussed above. Similarly, column 8, lines 6-9 of Wang state:

> The Network Protocol 46 software component on the network I/O device 40 is responsible for creating network layer data packets using the raw file data and sends the data to Network Attached Clients 12.

Moreover, applicants respectfully disagree with the Office Action statement that "Willis teaches that UDP, by definition, prepends data with UDP headers, wherein the data is further divided into plural fragments which are concatenated corresponding to the UDP header (col. 10, line 40 – col. 11, line 22; col. 14, lines 12-67; col. 15, lines 34-46)." Applicants have reviewed the cited passages of Willis and cannot find the teaching to which the Office Action refers.

Applicants respectfully assert that, *assuming arguendo* that one of ordinary skill in the art would have combined Wang and Willis as proposed by the Office Action, one of such skill may have used "planar board 20 having main CPU(s) 24 and main dynamic memory 22" of Wang to prepend a UDP header, instead of a "network I/O device," in contrast to claim 5. For these additional reasons, claim 5 is nonobvious over the references cited in the Office Action.

9. Claim 6

Regarding claim 6, the rejection of claims 4, 5 and 6 is shown above. Claim 6, however, recites:

> The apparatus of claim 1, wherein said data is disposed in plural fragments, and said interface device includes a mechanism configured to concatenate said fragments corresponding to a User Datagram Protocol header.

Applicants respectfully assert that, *assuming arguendo* that one of ordinary skill in the art would have combined Wang and Willis as proposed by the Office Action, the "network I/O device," would not have included "a mechanism configured to concatenate said fragments corresponding to a User Datagram Protocol header," in contrast to claim 6. Wang does not specifically teach how to receive packets from a network, instead stating in column 9, lines 10-17, that:

> The initial implementation of Peer I/O Manager 26 is designed to expedite file read operations, because it is believed that read operations are perceived as more time critical than write operations. This is true for applications such as video playback. However, the basic preferred implementation is fully capable of performing file write operations through Peer I/O.

Applicants respectfully assert, however, that write operations may be much more complicated, because the data can be out of order or erroneous when received from the network, unlike read operation data that are under complete control of the host computer. That is, parsing, sorting, validation, reassembly of out of order segments, etc. are not issues for sending but are issues for receiving data. For this reason, *assuming arguendo* that Wang and Willis would have been combined as proposed by the Office Action, the resulting combination would not teach that "said interface device includes a mechanism configured to concatenate said fragments corresponding to a User Datagram Protocol header," in contrast to claim 6.

Even for the simpler case of sending data, however, Wang makes clear that only network layer protocols, not transport layer protocols such as UDP, are handled by its "network I/O device." For example, according to column 12, lines 57-63 of Wang:

> In response to file read requests, the Peer I/O Manager transmits file data to the client redirector extension-based clients via raw IPX data packets. This means that no higher layer protocol information is contained in the IPX--Data field. The client redirector extension relies upon the SocketID in the IPX Header to properly assemble and process the received data.

For this additional reason, *assuming arguendo* that Wang would have been modified as proposed in the Office Action regarding claim 1, the resulting combination would not teach that "said interface device includes a mechanism configured to concatenate said fragments corresponding to a User Datagram Protocol header," in contrast to claim 6. Moreover, applicants respectfully disagree with the Office Action assertion that: "Willis teaches that UDP, by definition, prepends data with UDP headers, wherein the data is further divided into plural fragments which are concatenated corresponding to the UDP header (col. 10, line 40 – col. 11, line 22; col. 14, lines 12-67; col. 15, lines 34-46)." Applicants have reviewed the cited passages and respectfully disagree that "Willis teaches... plural fragments which are concatenated corresponding to the UDP header." For these additional reasons, claim 6 is nonobvious over the references cited in the Office Action.

10. Claim 7

Regarding claim 7, the Office Action states:

> As for claim 7, Wang teaches the apparatus of claim 1, wherein said data does not enter said host computer (col. 3, lines 31-52; Fig. 3).

Applicants respectfully assert that, *assuming arguendo* that Wang and Willis would have been combined as proposed in the Office Action, the resulting combination would not operate such that "said data does not enter said host computer," in contrast to claim 7. This is because, as discussed above, Wang teaches that "The client redirector extension relies upon the SocketID in the IPX Header to properly assemble and process the received data," yet the combination of Wang and Willis proposed by the Office Action would presumably replace the "unique Socket ID" of Wang's "Peer I/O Manager" with Willis' "UDP socket (UDP socket 5A30, fig. 5; UDP socket 6A20, fig. 6; col. 14, lines 12-67)." Applicants respectfully assert that with the "unique Socket ID" of Wang's "Peer I/O Manager" replaced, Wang would not operate such that "said data does not enter said host computer," in contrast to claim 7. For these additional reasons, claim 7 is nonobvious over the references cited in the Office Action.

C. The Rejection of Claims 11 and 13-18

Claims 11 and 13-18 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,913,028 to Wang et al. in view of U.S. Patent No. 6,385,647 to Willis et al. The Office Action states:

> Claims 11 and 13-18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Wang et al. (US 5,913,028) (hereinafter Wang) in view of Willis et al. (US 6,385,647) hereinafter Willis.
>
> As for claim 11, it is rejected for the same reasons set forth in claim 1 above, In addition, Wang discloses a host computer having a processor (CPU 24, Fig. 2) connected to a host memory (memory 32, Fig. 2) by a host memory bus (connection illustrated in Fig. 2), said host memory containing an application operable by the processor to designate a *socket* (col. 4, line 38 – col. 5, line 5; Fig. 3), and
>
> an interface device (network I/O device 40, Fig. 2) connected to said host computer and coupled between the network and the peripheral device, said interface device including an interface memory adapted to store data corresponding to *said socket* and a mechanism configured to associate said data with *a header* corresponding to *said socket* such that said data is communicated between the network and the peripheral device without encountering said host computer (col. 4, line 38 – col. 5, line 5; Fig. 3; bypassing, col. 17, lines 58-63).

D. Applicants' Response to the Rejection of Claims 11 and 13-18

1. Wang and Willis would not have been Combined by One of Ordinary Skill in the Art as Proposed by the Office Action

As mentioned above, applicants respectfully assert that Wang's disclosure of data transfer between Wang's "storage I/O device" and "network I/O device" would be destroyed if combined with Willis' UDP implementation as proposed by the Office Action. This is because Wang's "Peer I/O Manager," which appears essential to data transfer between Wang's "storage I/O device" and "network I/O device" is a protocol layer with a "unique Socket ID" that would be rendered ineffectual if replaced with Willis' UDP socket ID as proposed by the Office Action. As stated in column 11, lines 6-21 of Wang, which is cited in the Office Action for teaching a "*socket*":

> The IPX software layer acts much like a mailman. It delivers packets to higher layer protocols through a post office box-like scheme. The IPX layer uses the ***Destination Socket*** in the IPX Header field of each incoming packet as the postal address that associates incoming data packets with specific higher layer protocols. Protocols register with IPX to receive packets addressed to specific ***Socket IDs***.

> FIG. 7, illustrates the NetWare™ Core Protocol (NCP) as a software layer above IPX. IPX delivers data packets addressed to ***Socket number 451 to NCP***. Traditionally, NetWare™ client-based redirectors send file requests to ***the NCP socket ID*** in the file server.
>
> ***The Peer I/O Manager***, like NCP, is a ***protocol layer*** that accepts data packets addressed to ***its own unique Socket ID*** in a file server. The client redirector extension is a client-based software protocol layer that addresses file requests to the Peer I/O Manager socket ID in the file server. (emphasis added)

As further stated in column 4, lines 1-5 of Wang:

> The Peer I/O Manager implementation provides the program control necessary for coordinating and initiating peer I/O transfers directly from the storage I/O device to the network I/O device.

In other words, "using the Willis' UDP socket" in place of the "the Peer I/O Manager" as proposed by the Office Action would appear to destroy "the program control necessary for coordinating and initiating peer I/O transfers directly from the storage I/O device to the network I/O device." Applicants respectfully assert that for this reason one of ordinary skill in the art would not have combined Wang and Willis as proposed by the Office Action. For at least this reason, claim 11 is nonobvious over the Office Action's proposed combination of Wang and Willis.

2. *Assuming Arguendo* that Wang and Willis would have been Combined by One of Ordinary Skill in the Art as Proposed by the Office Action, Claim 11 would have Nonobvious Differences over that Combination

As mentioned above, applicants respectfully assert that if Wang was combined with Willis as proposed by the Office Action, Wang would not transfer data between Wang's "storage I/O device" and "network I/O device." Stated differently, the limitation of claim 11 "that said data is communicated between the network and the peripheral device without encountering said host computer" is very different than the Office Action's proposed combination of Wang and Willis, which would not perform this function. For at least this reason, claim 11 is nonobvious over the Office Action's proposed combination of Wang and Willis.

3. The Amendments to Claim 11

Although applicants assert that claim 11 as originally presented is nonobvious over the combination of Wang and Willis proposed by the Office Action as discussed above, in an effort to expedite prosecution applicants have amended claim 11 as discussed below. Applicants do not believe that claim 11 is obvious over the cited references without this amendment, and intend to file a continuation application that includes a claim having the limitations of claim 11.

In the present application, however, applicants have amended claim 11 to recite "a host computer having a processor running a file system and ... an interface file cache managed by said file system." Applicants respectfully assert that these limitations of claim 11 are nonobvious over the combination of Wang, Gentry and Willis much as discussed above with regard to claim 1.

4. Claim 13

Regarding claims 13 and 14, the Office Action states:

> As for claims 13 and 14, Wang does not explicitly disclose the use of UDP packets and headers. Stevens teaches that UDP, by definition, includes UDP packets and headers, wherein said data travels over the network in plural fragments (packets) corresponding to the header. It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Wang by using UDP packets and headers, wherein the interface device processes the headers and concatenates the data, because these are well-known and necessary steps in order to efficiently transfer data over a network, as taught by Willis above.

Claim 13, however, recites:

> The apparatus of claim 11, wherein said data travels over the network in at least one packet containing a User Datagram Protocol header, and said interface device includes circuitry configured to process said User Datagram Protocol header.

Applicants are unsure what is meant by the Office Action reference to Stevens, because the Office Action states that this rejection is over the combination of Wang and Willis. Applicants assume, however, that the Office Action is referring to "Stevens, TCP/IP Illustrated, Volume 1: The Protocols," New York, 1994, which was cited in a previous Office Action. There is no teaching or suggestion in Wang, Willis or Stevens

that "said interface device includes circuitry configured to process said User Datagram Protocol header," as recited in claim 13. Wang does not even mention the words "circuit" or "circuitry," and such circuitry is not inherent in Wang. The Office Action also does not assert that Wang or any other cited reference teaches such circuitry, and for at least these reasons does not present a *prima facie* case of obviousness of claim 13.

5. Claim 14

Similarly, claim 14 recites:

> The apparatus of claim 11, wherein said data travels over the network in plural fragments corresponding to a User Datagram Protocol header, and said interface device is configured to concatenate said data with said User Datagram Protocol header.

Applicants respectfully assert that there is no teaching or suggestion in Wang, Willis or Stevens that "said interface device is configured to concatenate said data with said User Datagram Protocol header," as recited in claim 14. Applicants respectfully disagree with the Office Action assertion that: "It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Wang by using UDP packets and headers, wherein the interface device processes the headers and concatenates the data, because these are well-known and necessary steps in order to efficiently transfer data over a network, as taught by Willis above." Applicants respectfully disagree with the Office Action assertion that Willis teaches that concatenating UDP data is a well-known and necessary step in order to efficiently transfer data over a network. Willis does not mention concatenating data. Applicants respectfully disagree with the Office Action assertion that processing UDP packets and concatenating data by an interface device are well known and necessary steps, and respectfully request the Examiner to provide an affidavit supporting his assertion as required by 37 CFR 1.104(d)(2).

6. Claim 15

Regarding claim 15, the Office Action states:

> As for claim 15, Wang does not explicitly disclose the use of Realtime Transport Protocol (RTP). Willis teaches the use of RTP and RTP headers in order to transfer data over a network while maintaining the real-time characteristics (col. 11, lines 2-22). It would have been obvious

to one of ordinary skill in the art at the time of the invention to combine the teachings of Wang and Willis because Willis' RTP protocol would improve the quality of service by supporting the transmission and reception of real-time multimedia (willis, col. 2, lines 26-36).

Claim 15, however, recites:

> The apparatus of claim 11, wherein said host computer is configured to create a Realtime Transport Protocol header that is accessible by said interface device, and said interface device is configured to prepend said Realtime Transport Protocol header to said data.

Applicants respectfully assert that, as discussed above, Wang's disclosure of data transfer between Wang's "storage I/O device" and "network I/O device" would be destroyed if combined with Willis' UDP implementation as proposed by the Office Action. This is because Wang's "Peer I/O Manager," which appears essential to data transfer between Wang's "storage I/O device" and "network I/O device" is a protocol layer with a "unique Socket ID" that would be rendered ineffectual if replaced with Willis' UDP socket ID as proposed by the Office Action. For this reason, one of ordinary skill in the art would not have combined Wang and Willis as proposed in the Office Action.

Moreover, applicants respectfully assert that none of the cited references teaches or suggests that "said host computer is configured to create a Realtime Transport Protocol header" and that "said interface device is configured to prepend said Realtime Transport Protocol header to said data." Nor does the Office Action assert that this would somehow result, *assuming arguendo* that Wang and Willis would have been combined as proposed in the Office Action.

Applicants respectfully assert that, *assuming arguendo* that one of ordinary skill in the art would have modified Wang to use RTP as proposed by the Office Action with regard to claim 11, one of such skill may have instead have used a host computer to prepend a RTP header, or used an interface device to create a RTP header, in contrast to claim 15. For these additional reasons, claim 15 is nonobvious over the references cited in the Office Action.

E. The Rejection of Claims 2 and 22

Claims 2 and 22 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,913,028 to Wang et al. in view of U.S. Patent No. 5,848,293 to Gentry and U.S. Patent No. 6,385,647 to Willis et al. and "Applicant's admitted prior art." The Office Action states:

> **Claims 2 and 22** are rejected under 35 U.S.C. 103(a) as being unpatentable over Wang in view of Gentry, Willis and further in view of Applicant's admitted prior art (pg. 31, line 28 – pg. 32, line 8) (hereinafter AAPA).
>
> As for claims 2 and 22, Wang explicitly teaches the use of application layer headers, which inherently includes prepending these headers to the data (col. 11, lines 14-36), because otherwise the data packets could not be transferred. It is not clear from Wang whether or not these application layer headers are created by the host computer or the interface device. Thus, Wang, Gentry and Willis do not specifically disclose a host computer that is configured to create an application layer header that is accessible by said interface device. However, AAPA (pg. 31, line 28 – pg. 32, line 8) teaches that it is conventionally known to generate an application packet header at a host computer. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Wang Gentry and Willis by configuring the host computer to create an application layer header that is accessible by said interface device in order to transmit application data over the network. Moreover, it would have been obvious to generate the application layer headers at the host computer of Wang, because the application resides on the host computer and this would simplify data processing.

E. Applicants' Response to the Rejection of Claims 2 and 22

1. Claim 2

Claim 2 recites:

> The apparatus of claim 1, wherein said host computer is configured to create an application layer header that is accessible by said interface device, and said interface device is configured to prepend said application layer header to said data.

Regarding claim 2, the Office Action fails to show the existence of any incentive in the cited references for why "it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the teachings of Wang Gentry and Willis by configuring the host computer to create an application layer header that is accessible by

said interface device in order to transmit application data over the network." Instead, the Office Action admits that "Wang, Stevens and Willis do not specifically disclose a host computer that is configured to create an application layer header that is accessible by said interface device." The passage from applicants' specification that is cited by the Office Action does not state that the prior art includes such a motivation. For at least these reasons claim 2 is not obvious over the cited references.

In addition, the Office Action does not show that other limitations of claim 2, for example that "said interface device is configured to prepend said application layer header to said data," are taught or suggested by the cited references. In addition, no motivation to modify the cited references to include such other limitations is provided by the Office Action.

For at least these various reasons, the Office Action fails to state a *prima facie* case of obviousness of claim 2.

B. Claim 22

Claim 22 recites:

> The apparatus of claim 21, wherein said host computer is configured to create an application layer header that is accessible by said interface device, and said interface device is configured to prepend said application layer header to said data.

The Office Action does not distinguish claim 22 from claim 2, despite the "means-plus-function" clause in claim 21, from which claim 22 depends. Because the Examiner does not attempt to show that the structure corresponding to this clause in claim 21 is obvious over the cited references as implicated in claim 22, the Office Action has failed to present a *prima facie* case of obviousness of claim 22.

In addition, applicants incorporate by reference the arguments made above with regard to claim 2, to further explain why claim 22 is not obvious over the cited references.

Conclusion

As detailed above, applicants believe that all the pending claims are allowable and respectfully request a Notice of Allowance.

A Petition for an Extension of Time, a Terminal Disclaimer and an Information Disclosure Statement are enclosed. Also enclosed is a check in the amount of $430.00 to pay the Extension of Time fee ($120.00), the Terminal Disclaimer fee ($130.00), and the IDS fee ($180.00).

Respectfully submitted,

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to Mail Stop Amendment, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on February 27, 2006.

Date: 2-26-06

Mark Lauer

Mark Lauer
Reg. No. 36,578
6601 Koll Center Parkway
Suite 245
Pleasanton, CA 94566
Tel: (925) 484-9295
Fax: (925) 484-9291